

1st Seminar on Application and Research in Industrial Technology, SMART 2006
Yogyakarta, 27 April 2006

Metode Mengatasi Keragaman Semantik antara Informasi di Pemasok dan Manufaktur pada Manajemen Rantai Persediaan

Lily Wulandari dan I Wayan Simri Wicaksana

Universitas Gunadarma, Jl. Margonda Raya no.100, Depok

Email: lily@staff.gunadarma.ac.id dan iwayan@staff.gunadarma.ac.id

Abstrak

Pemanfaatan teknologi informasi semakin tinggi pada dunia manufaktur. Sehingga banyak aktifitas produksi dari awal hingga akhir tergantung pada teknologi informasi. Internet membawa dampak pada keragaman informasi dari pemasok dan pemakai pada berbagai hal seperti format data, penyajian informasi, representasi informasi, dan pengaksesan. Pada kegiatan rantai persediaan, salah satu langkah awal adalah untuk menemukan pemasok yang sesuai dengan kebutuhan sebuah manufaktur. Tetapi keragaman yang terjadi menimbulkan permasalahan untuk memahami antara permintaan dan penawaran, seperti istilah 'penyimpanan' – 'RAM' – 'gudang' – 'kontainer'.

Permasalahan ini memotivasi pendekatan semantic web dan ontologi ke domain manufaktur pada manajemen rantai persediaan, khususnya dalam pencarian informasi untuk kesesuaian antara pemasok dan manufaktur.

Semantic web adalah mendorong 'mesin' yang lebih mampu memahami informasi untuk digunakan dalam pencarian informasi yang lebih baik. Ontologi akan diterapkan sebagai sarana utama untuk mengatasi keragaman untuk mencapai interoperabilitas. Interoperabilitas semantik dicapai dengan hubungan antar terminologi ke ontologi, seperti menggunakan model bahasa yang diperkaya dengan struktur internal dan eksternal.

Pendekatan diatas akan dikembangkan menjadi prototipe untuk interoperabilitas dengan Semantic Web dan ontologi menggunakan OWL yang berbasis pada RDF/S dan XML/S. Tools utama untuk pengembangan ontologi dan representasi sumber adalah memanfaatkan Protégé. Simulasi akan memanfaatkan virtual server (Apache) dan virtual machine (QEMU).

Kata kunci: manajemen rantai persediaan, interoperabilitas informasi, keragaman informasi, semantic web, ontology

1. Pendahuluan

a. Latar Belakang

Pemanfaatan teknologi informasi semakin tinggi pada dunia manufaktur. Sehingga banyak aktifitas produksi dari awal hingga akhir tergantung pada teknologi informasi. Internet membawa dampak pada keragaman informasi dari pemasok dan pemakai pada berbagai hal seperti format data, penyajian informasi, representasi informasi, dan pengaksesan. Pada kegiatan rantai persediaan, salah satu langkah awal adalah untuk menemukan pemasok yang sesuai dengan kebutuhan sebuah manufaktur. Tetapi keragaman yang terjadi menimbulkan permasalahan untuk memahami antara permintaan dan penawaran, seperti istilah 'penyimpanan' – 'RAM' – 'gudang' – 'kontainer'.

b. Related Work

Pada bagian ini, beberapa pendekatan dari interoperabilitas semantik diperkenalkan. Pendekatan dapat dibagi menjadi dua pendekatan yang umum: *tightly solution* (bagan terintegrasi secara global dan ontologi yang terintegrasi secara global) dan *loosely couple* (menggunakan mediasi). Pendekatan mediasi lebih sesuai untuk memecahkan karakteristik kondisi saat ini (lingkungan heterogen, yang dinamis, peserta/pemain dalam jumlah yang sangat besar, dan lain-lain).

Paradigma Interoperabilitas ditimbulkan dengan isu pemetaan. Saat ini, proses pemetaan dan menggabungkan mempunyai tantangan yang besar dalam hal otomatisasi secara penuh. Semantik Mediasi

untuk interoperabilitas menjanjikan pemecahan kondisi untuk interoperabilitas saat ini. Walaupun, ada beberapa permasalahan [Richard Benjamins, et. al, 2002], Semantik Mediasi dapat dikategorikan sebagai berikut:

- *The Availability of content*: implementasi semantik untuk menguraikan konten/isi dari WWW adalah sangat terbatas.
- *Scalability*: pertumbuhan halaman web akan mendorong pertumbuhan Semantic Web, metoda untuk menangani pertumbuhan secara *scalable* adalah penting.
- *Multilinguality* pada level mesin dan manusia, ada banyak bahasa untuk menciptakan Semantic Web, dan juga ada banyak bahasa untuk manusia.
- Stabilitas dari bahasa Semantic Web, untuk mendukung Semantic Web, usaha standardisasi masih penting untuk dikembangkan.
- Ketersediaan, pengembangan dan evolusi ontologi, ontologi adalah alat penting untuk Semantic Web. Bagaimanapun, Maturity dalam pengembangan, dan pemeliharaan kebutuhan ontologi memerlukan cara yang lebih baik.
- Visualisasi, meningkatkan informasi dalam jumlah yang berlebih, pemakai akan menuntut pendekatan yang baru yang berbeda dari hypertext yang umum dalam visualisasi untuk mendukung pemakai.

c. Kontribusi dan Pembagian Tulisan

Pada industri yang fleksibel / *craft*, ukuran volume produk adalah dari level kecil ke menengah. Perubahan akan jenis produk juga membawa ke perubahan akan kebutuhan material. Dalam mencari material diperlukan akses ke sumber informasi dari pemasok. Disisi lain terjadinya keragaman informasi dari pemasok dalam format data, penyajian informasi, representasi informasi, dan juga mengaksesnya. Permasalahan ini memotivasi kami membawa pendekatan semantic web dan ontologi ke domain manufaktur untuk jaringan produksi, khususnya dalam pencarian informasi untuk *raw material*.

Paper ini akan dibagi menjadi beberapa bagian, pada bagian 2 yaitu tentang *Semantic Mediation* akan melihat secara ringkas tentang pemahaman hybrid Ontology dan Semantic Web serta membahas mengenai contoh kasus dan perhitungan similaritas dengan *Label Matching*, internal struktur dan eksternal struktur. Bagian 3 akan menguraikan rencana prototipe. Bagian 4 adalah kesimpulan dari paper kami.

2. Semantic Mediation

a. Hybrid Ontology dan Semantic Web

Pendekatan ontologi dan semantik web adalah sebuah pendekatan relatif baru untuk sistem interoperabilitas yang menggabungkan pendekatan dari sisi database dan intelegensi buatan. Berbagai perbedaan dalam persepsi informasi dari satu perusahaan ke yang lainnya bisa sekali terjadi. Dalam paper ini akan dikemukakan sebuah perusahaan perakitan komputer yang bersifat 'craft industry'. Sebagai motivasi permasalahan adalah banyaknya berbagai terminologi dan klasifikasi yang dinamis di industri komputer. Sebagai contoh sederhana, kalau kita bicara komputer server kelas menengah, maka pemahaman dari kelas menengah dari satu perusahaan ke yang lainnya bisa berbeda. Sehingga diperlukan sebuah pendekatan untuk interoperabilitas informasi terutama pada bagian pembelian 'raw material' untuk perakitan yang sesuai.

1) Semantic Web

Semantic Web telah mengantar evolusi WWW ke tingkat pemanfaatan yang lebih baik. Ada dua visi dalam pengembangan web ke depan, yaitu pertama, membuat web semakin baik sebagai media kolaborasi, dan yang kedua, web semakin dapat dipahami oleh mesin. Dengan memberikan anotasi data akan membuat informasi yang lebih dapat dipahami oleh mesin. Untuk mengembangkan Semantic Web [Berners-Lee, 1999] beberapa hal dibutuhkan seperti :

- Mengembangkan bahasa dan terminology. Bahasa yang digunakan untuk mengkespresikan sesuatu yang dapat membuat mesin lebih bisa memahami dengan meta-informasi untuk dokumen, dan terminology sebagaimana ontologi digunakan untuk mengekspresikan konsistensi dari semantic.
- Mengembangkan tool dan arsitektur baru yang menggunakan bahasa dan terminology tersebut untuk mengakses, merubah dan integrasi informasi.
- Mengembangkan aplikasi yang memberikan sebuah tingkat pelayanan baru kepada pemakai dengan Semantic Web.

Model yang akan digunakan untuk interoperabilitas dengan Semantic Web dan ontologi sebagai penunjang jaringan sistem produksi akan dimulai dengan lapis 1 (XML, XMLS) dan lapis 2 (RDF,

RDFS). XML adalah sebenarnya bukan sebuah bahasa, lebih sesuai dikatakan sebagai sebuah aturan sintaks untuk membuat bahasa markup yang lebih semantik pada domain tertentu. Dengan memberikan instruksi semantik, maka pemahaman sebuah informasi semakin mudah.

XML Schema (XMLS) adalah sebuah definisi dari XML untuk memberikan aturan untuk sebuah dokumen XML, seperti analoginya pada database konvensional adalah XMLS bagaikan data dictionary, sementara XML adalah data di Xbase-nya. Resources Description Framework (RDF) adalah sebagai sebuah bahasa formal yang berbasiskan XML, URI dan unicode. RDF adalah sebuah dasar untuk pemrosesan metadata, dimana metadata dalam web dapat di kodekan, dipertukaran dan dipergunakan. RDF terdiri dari tiga jenis bagian (triple) subyek-predikat-obyek, dimana bisa disebut juga sebagai:

- Resources, adalah bagian dari sumber informasi, dalam era Internet di representasikan dalam alamat web atau URL, ini disebut subyek atau obyek.
- Property, adalah sebuah karakteristik dari atribut atau relasi untuk menjelaskan sumber, ini disebut juga predikat.

RDF bukanlah melakukan pendefinisian semantik secara langsung dari setiap sumber, tetapi lebih melakukan penjelasan untuk lebih dapat dipahami oleh mesin, sehingga memudahkan untuk pertukaran data. RDF Schema (RDFS) adalah sebuah lapisan di atas RDF, dan merupakan sebuah set standard sederhana dari sumber RDF yang memungkinkan untuk membuat vokabulari RDF sendiri. Model dari RDFS memiliki kemiripan dengan yang digunakan oleh object-oriented, yaitu dengan memiliki class, relation, property dan instance. Class adalah kumpulan dari obyek yang memiliki kesamaan karakter. Relation adalah sifat hubungan antar kelas. *Property* adalah karakter dari sebuah class. Instances adalah sebuah obyek yang merupakan anggota sebuah class.

2) Ontologi

Sebuah ontologi adalah merupakan sesuatu yang formal, menjelaskan secara eksplisit sebuah konsep yang spesifik untuk saling berbagi pada domain khusus dalam sistem komputer (Gruber, 1995). Konsep ini mengacu kepada sesuatu yang abstrak, suatu pandangan dari hal nyata yang didefinisikan berelasi dengan konsep. Eksplisit artinya mendefinisikan tipe konsep dan batasan yang digunakan dalam memberikan arti formal yang dipahami oleh bahasa mesin.

Proses interoperabilitas atau integrasi umumnya ada tiga cara untuk mengatasi perbedaan semantik dalam memanfaatkan konsep dari ontologi. Model pertama adalah dengan membuat ontologi yang dipahami secara bersama oleh semua sumber informasi. Pendekatan ini sangat tersentral, dan cocok untuk sumber informasi yang statis dan relasinya sangat tinggi, sering diistilahkan dengan *merging ontology*. Model kedua adalah, setiap sumber informasi memiliki ontologi masing-masing, dan dalam proses interoperabilitas akan dilakukan mapping dari satu ontologi ke ontologi lainnya. Model ini mendukung otonomi dari sumber informasi, tetapi membutuhkan banyak mapping ontologi yaitu sebanyak $\sum(n-1)$, ini disebut model *mapping ontology*. Model ketiga, adalah dengan mengembangkan ontologi referensi / *common ontology*, sehingga pemetaan hanya dilakukan antara sumber dan ontologi referensi. Ini jelas sangat mengurangi jumlah mapping menjadi sebesar n , tetapi permasalahannya adalah bagaimana membangun ontologi referensi yang bisa menjadi acuan dari sebuah domain. Model ketiga ini disebut dengan model *ontology hybrid*, karena merupakan gabungan dari ide model ke satu dan ke dua.

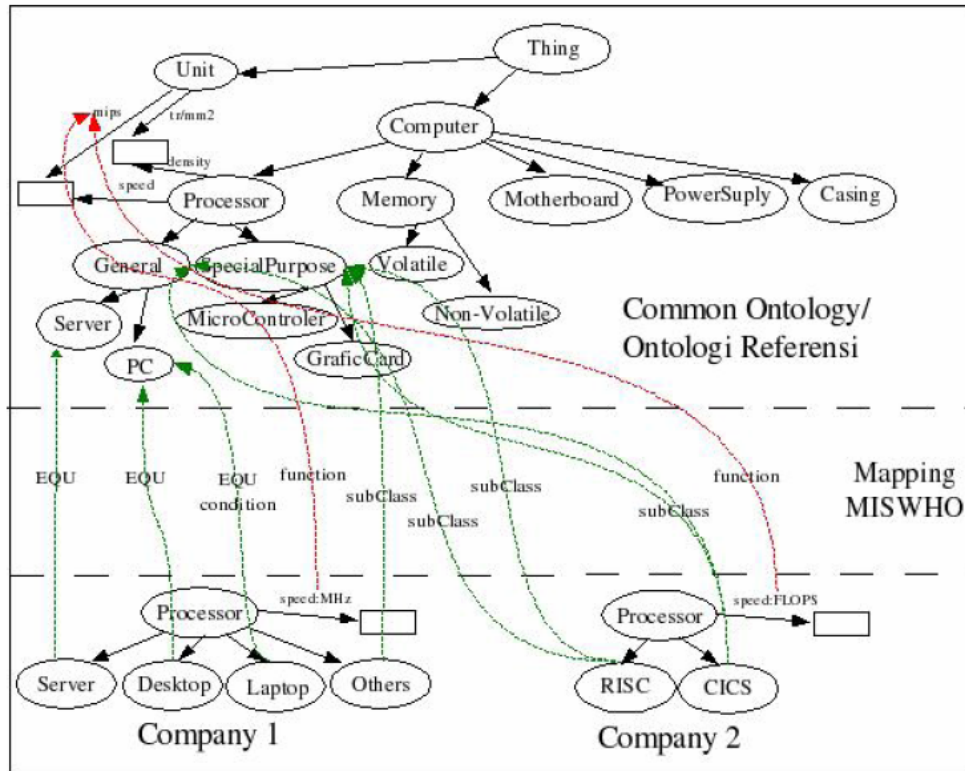
b. Contoh Kasus

Sebagai contoh sederhana adalah jika sebuah perusahaan perakitan PC yang bersifat khusus bukan masal membutuhkan sebuah prosesor khusus untuk memenuhi kebutuhan pelanggan. Perusahaan akan mencari sebuah processor untuk pengontrol industri yang memiliki kemampuan proses kelas menengah, ketahanan terhadap lingkungan kerja yang tidak terlalu ramah. Katakan perusahaan tersebut mencari dari Internet dan mengirimkan permintaan “membutuhkan prosesor khusus untuk sistem kontrol dengan kecepatan sedang”. Sebelum mengirim query tersebut, diperlukan langkah untuk query *rewriting* berbasiskan dari ontologi referensi. Hasil dari *rewriting* akan merubah query menjadi “membutuhkan Processor untuk MicroController dengan kecepatan 100-200 MIPS”. Perubahan yang terjadi adalah sistem kontrol menjadi MicroController, dan kecepatan sedang menjadi kecepatan 100-200 MIPS.

Masing-masing perusahaan (Company 1 dan Company 2) sebagai pemasok prosesor memiliki ontologi / skema lokal. Pada gambar 1, diperlihatkan model dari ontologi referensi yang disederhanakan serta ontologi lokal dari 2 perusahaan.

Query di atas akan direspon oleh sumber data di Company 1 dengan mengacu hasil MISWHO adalah query untuk *MicroController* akan diarahkan ke *Others*, dan besaran *MIPS* akan dikonversi ke *MHz* (ini dilakukan dengan *look-up table*, walau dalam praktis tidak mudah). Selanjutnya akan dicari dari class *Others* dengan processor kecepatan *X-Y MHz*. Jawaban dari Company 1 terhadap query akan terjadi beberapa *loss-information* atau *miss-information*, karena pemahaman dari *Others* lebih luas dari *SpecialPurpose* apalagi terhadap *MicroController*. Tetapi respon dari Company 1 dapat disaring lebih

lanjut oleh si pengirim query untuk melihat prosesor yang diinginkan. Ini jelas berbeda kalau kita menggunakan pencarian dengan metode *keywords* : “MicroController AND speed=100-200MIPS”, maka Company 1 dan 2 akan menjawab tidak memiliki data. Tetapi dengan pendekatan Semantic Web dan ontologi, dapat didalami sebuah sumber informasi untuk merespon query. Hal yang sama juga dilakukan kepada Company 2. Dari ilustrasi di atas, dapat dilihat bahwa dengan pendekatan ini akan memungkinkan untuk melakukan interoperabilitas yang lebih baik.



Gambar 1. Contoh Kasus untuk Ontologi Referensi, Lokal Ontologi dan MISWHO

c. Perhitungan Similarity

Istilah interoperabilitas semantik digunakan untuk memecahkan konflik semantik yang terjadi antar sumber informasi heterogen. Konflik semantik terjadi kapan saja dua sistem tidak menggunakan penafsiran informasi yang sama. Format konflik yang paling sederhana dalam penafsiran informasi adalah homonim (penggunaan kata yang sama yang memiliki arti yang berbeda), dan sinonim (penggunaan kata-kata yang berbeda yang memiliki arti yang sama).

Perhitungan similarity semantik antar konsep adalah langkah pertama untuk menciptakan persetujuan antar ontologi dan isi/konten dari penyedia atau yang meminta informasi. Masing-Masing konsep dapat diwakili sebagai hirarki menurut label yang berisi beberapa informasi semantik atau struktural. Berikut ini tiga pengukuran similarity [Wicaksana, IWS, 2006] :

- **Label Matching**

Proses berdasarkan pada analisa ilmu bahasa. Ada dua teknik umum pada label matching [Fausto Giunchiglia, et.al, 2003]. Pertama, suatu langkah analisis linguistik, seperti merubah singkatan, menghindari perulangan, imbuhan-akhiran. Berikutnya, label dicocokkan dengan menentukan hubungan antar mereka. Ini dilaksanakan oleh teknik yang menggunakan thesauri atau tool wordnet. Pendekatan ini intinya menghitung kesesuaian struktural antar label berdasarkan pada arti kata yang tersembunyi yang disampaikan oleh label. Hasil perhitungan ini dapat dinyatakan dalam tuple $\langle LCO_i, LPP_j, SimLabel \rangle$, dimana LCO_i adalah label ke- i pada CO(industri perakitan), LPP_j adalah label ke- j pada PP (pemasok), $SimLabel$ adalah perhitungan similaritas berdasarkan WordNet. Hasil dari langkah pertama diperkaya dengan pendekatan perbandingan struktur internal dan eksternal.

- **Internal Struktur**

Similaritas antara dua konsep dapat diperoleh dengan membandingkan 'language' dan 'real' attribute dan tidak hanya dari persamaan antara uraian komponen mereka, tetapi juga dari persamaan antar struktur dari grafik yang mewakili mereka. Similaritas dari struktur internal dapat diperoleh dengan menghitung banyaknya attribute yang serupa dibagi jumlah maksimal attribute dari sebuah kelas. $IS = \text{similar attribute} / [\text{maks attribute pada sebuah kelas}]$. Hasil ini juga dinyatakan dengan tuple $\langle CCO_i, CPP_j, SimIS \rangle$, dimana CCO_i adalah class ke- i pada CO (dalam hal ini industri perakitan), CPP_j -k adalah class ke- j pada PP (pada pemasok), $SimIS$ adalah perhitungan perbandingan internal struktur.

- **Eksternal Struktur**

Perbandingan eksternal struktur adalah melihat set dari upper-class. Secara sederhana untuk menghitung eksternal struktur dari dua kelas adalah dengan melihat berapa jumlah upper-class yang sama akan dibagi dengan jumlah upper-class terbesar dari sebuah kelas [Jayant Madhavan, et.al, 2001]. $ES = \text{similar upper-class} / [\text{maks upper-class pada sebuah kelas}]$. Hasil ini juga dinyatakan dengan tuple $\langle CCO_i, CPP_j, SimES \rangle$, dimana CCO_i adalah class ke- i pada CO (industri perakitan), CPP_j -k adalah class ke- j pada PP (pemasok), $SimES$ adalah perhitungan perbandingan eksternal struktur.

Melihat kepada contoh kasus, maka kita dapat melihat menghitung berbagai kelas dari Company1, Company2 terhadap Common Ontologi.

- Label Matching $\langle LCO_i, LPP_j, SimLabel \rangle$, menggunakan persamaan dari *path length* :
 - $\langle CO: \text{Thing-Computer-Processor-General-PC}, Company1: \text{Processor-Laptop}, 0.20 \rangle$
 - $\langle CO: \text{Thing-Computer-Processor-General-PC}, Company2: \text{Processor-RISC}, 0.07 \rangle$
- Internal Struktur $\langle CCO_i, CPP_j, SimIS \rangle$
 - $\langle CO: \text{Thing-Computer-Processor-General-PC}[\text{name, clock, branded}], Company1: \text{Processor-Laptop}[\text{name, speed, year, bit}], 2/4 \rangle$
 - $\langle CO: \text{Thing-Computer-Processor-General-PC}[\text{name, clock, branded}], Company2: \text{Processor-RISC}[\text{name, clock, branded}], 3/3 \rangle$
- Eksternal Struktur $\langle CCO_i, CPP_j, SimES \rangle$
 - $\langle CO: \text{Thing-Computer-Processor-General-PC}, Company1: \text{Processor-Laptop}, 1/4 \rangle$
 - $\langle CO: \text{Thing-Computer-Processor-General-PC}, Company2: \text{Processor-RISC}, 1/4 \rangle$

Berdasarkan hasil di atas untuk mencari informasi processor laptop terhadap processor RISC, maka secara linguistik Laptop lebih baik dibandingkan RISC. Secara internal struktur, RISC lebih baik dari pada Laptop, secara eksternal struktur memiliki nilai yang sama. Untuk menggabungkan hasil di atas maka perlu diadakan pembobotan dari masing-masing pendekatan. Sebagai contoh misalkan kita menggunakan pembobotan untuk label matching 70%, internal struktur 10% dan eksternal struktur 20%. Maka hasil yang di dapat adalah :

$$\text{Laptop} = 70\% \times 0.20 + 10\% \times (2/4) + 20\% \times (1/4) = 0.240$$

$$\text{RISC} = 70\% \times 0.07 + 10\% \times (3/3) + 20\% \times (1/4) = 0.199$$

Melihat hasil di atas, maka Laptop memiliki kedekatan yang lebih baik ke Processor utk PC dibandingkan dengan RISC. Hal ini dapat saja tidak benar, karena representasi ontologi lokal/taksonomi tidak sesuai. Sehingga dalam hal ini representasi pengetahuan atau skema atau taksonomi sebuah data maupun informasi menjadi sangat penting dalam pertukaran informasi.

3. Rancangan Prototype

Arsitektur dari model ini dapat dikembangkan dengan mengadopsi dari berbagai model sebagai berikut:

- Menggunakan loosely couple arsitektur, yang dapat dimanfaatkan pada bentuk client server atau hybrid peer-to-peer (dengan super peer).
- Pemanfaatan ontologi dengan pendekatan sebagai referensi dengan menggunakan common / share ontologi.
- Mediasi model juga dimanfaatkan untuk pertukaran antara berbagai sumber informasi.

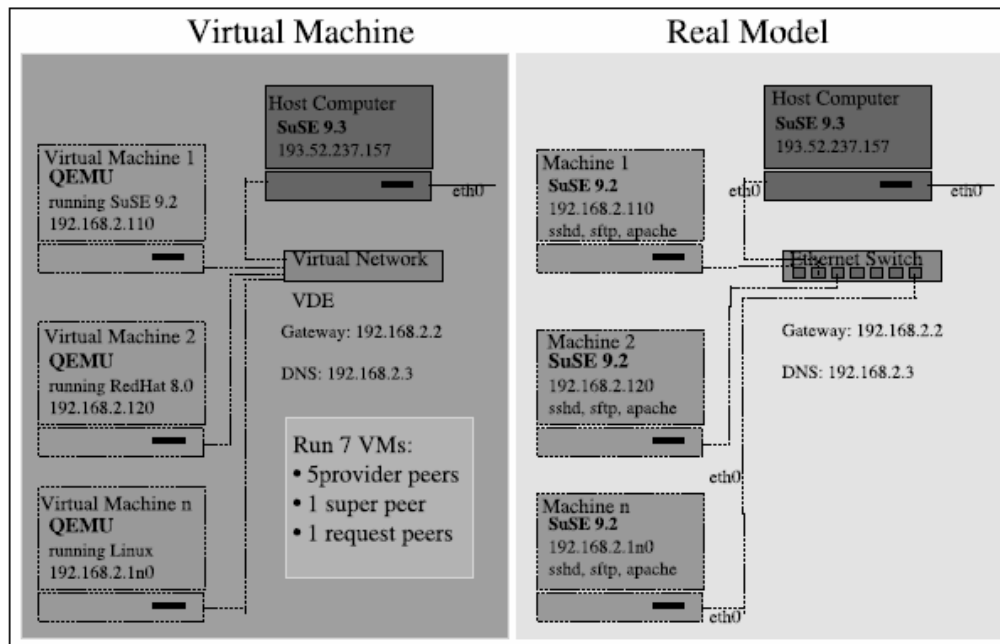
Komponen dari sistem adalah sebagai berikut :

Server atau Super Peer akan memiliki fungsi untuk (1). menyediakan dan memelihara common ontologi. (2) mencatat berbagai sumber informasi beserta metadata dari informasi yang tersedia, dan (3) menginformasikan server atau super peer lain untuk menghindari 'single point failure'

Sumber informasi akan memiliki fungsi untuk (1) menyajikan data atau informasi yang dapat dipergunakan oleh pihak eksternal, (2) menyajika informasi dari skema atau ontologi lokal, (3) membuat

mapping antara ontologi lokal dan di server/super peer, (4) merespond query dari pengguna, dan (5) mekanisme registrasi ke server/super-peer termasuk untuk memberitahukan kondisi aktif/tidak dan juga jika adanya perubahan data atau konsep.

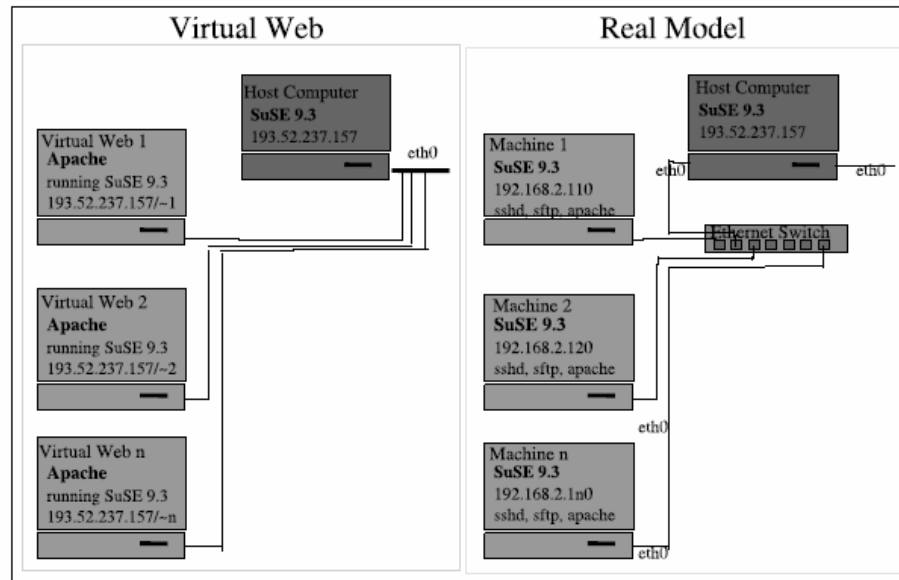
Secara sederhana bentuk prototipe untuk pengujian dasar dapat dilihat pada gambar 4 dan 5.



Gambar 2. Virtual Machine Model untuk Prototipe

Untuk uji coba awal pendekatan di atas, kami membuat prototipe sederhana dengan berbasiskan pada sistem operasi SuSE 9.3 dan menggunakan tool Protégé 3.2.beta, serta memanfaatkan virtual mesin dari QEMU dan virtual web server dari Apache.

Pada percobaan kami membangun common dan lokal ontologi dengan jumlah class masing-masing sumber di bawah 10, dengan instances 10 buah juga. Hal ini dilakukan karena masih merupakan inisial test terhadap prototipe kami. Kemudian setelah terbentuknya common dan lokal ontologi, kami melakukan mapping secara manual dengan dibantu pendekatan similaritas di atas. Hasil dari mapping kembali dimasukkan dalam bentuk OWL menggunakan tool Protégé. Masing-masing sumber informasi kami lakukan simulasi dengan dua model, dengan menggunakan QEMU, yang akan memberikan kondisi seperti benar-benar terpisah secara fisik. Dan juga menggunakan virtual web-server untuk memisahkan secara logik.



Gambar 3. Virtual Web Model untuk Prototipe

Untuk menguji prototipe kami, kami hanya menekankan pada kemampuan sumber informasi untuk menangani query yang dikirim. Ini akan memperlihatkan hasil dari similaritas untuk membuat mapping. Pengujian dengan mengirim query untuk mendapatkan class ataupun instance ke berbagai sumber. Kemudian kami juga melakukan kondisi dinamis, dimana merubah konsep dan ketersediaan sumber informasi.

Dari percobaan dengan kasus sederhana di atas, maka pendekatan yang kami lakukan adalah dapat memberikan hasil yang memadai dengan bisa membedakan mana informasi yang sesuai atau tidak. Unjuk kerja dari QEMU sangat lambat dibandingkan dengan Apache, dan hasil respon query adalah sama.

4. Penutup

Pemanfaatan ontologi, penghitungan similaritas, semantik dan arsitektur yang tepat, dapat memberikan kemampuan akses informasi pada beragam sumber informasi dimana sumber informasi tadi bisa bersifat otonomi dan dinamis. Pada paper ini kami telah memberikan kontribusi dengan pemanfaatan similaritas semantik untuk memperbaiki akses data dan informasi dalam manajemen rantai persediaan.

Pada langkah ke depan kami akan mengembangkan prototipe dan pengujian yang lebih melihat kepada kondisi nyata dengan beban jumlah class dan instance yang jauh lebih besar, termasuk juga keragaman dan jumlah sumber informasi.

Daftar Pustaka

- Berners-Lee, T., 1999, Weaving the Web, The Original Design and ultimate Destiny of the World Wide Web, Harper.
- Fausto Giunchiglia dan Pavel Shvaiko. April 2003, Semantic Matching. Technical Report DIT-03-013, University of TN.
- Fensel D., van Harmelen F., Maret 2001, OIL: an Ontology Infrastructure for the Semantic Web, *IEEE Intelligent Systems*, hal. 38-45.
- Jayant Madhavan, Philip A. Bernstein, dan Erhard Rahm., 2001, Generic Schema Matching with CUPID. In Proc. of 27th Intl. Conf. on Very Large Data Bases, Very Large Data Bases, hal. 49-58.
- Richard Benjamins, Jesus Contreras, Oscar Corcho, dan Asuncion Gomez-Perez, 2002, Six Challenges for The Semantic Web. KR2002 (ISOCO White Paper).
- Wicaksana, IWS, 2006, Desertasi Doktor : A Peer-to-Peer (P2P) Based Semantic Agreement Approach for Spatial Information Interoperability, Universitas Gunadarma.
- Wulandari, Lily dan Wicaksana, IWS, Agustus 2005, Semantic-Web Solusi Interoperabilitas Informasi Sebagai Penunjang Jaringan Sistem Produksi, Proc. Seminar Sistem Produksi VII '2005, ISSN:0854-431X, hal. 668-681
- _____, Januari 2005, WordNet homepage, <http://WordNet.princeton.edu>